

**SYSTEM AND METHOD FOR  
PROGRAMMING SYNCHRONIZATION CRITERIA  
IN A MULTIDIMENSIONAL DIGITAL FRAME STRUCTURE**

5 **RELATED APPLICATIONS**

This application contains material related to the following commonly assigned copending U.S. Patent Applications, incorporated herein by reference:

Serial No. \_\_\_\_\_ filed \_\_\_\_\_ for "SYSTEM  
10 AND METHOD FOR PROGRAMMING THE LOCATION OF FRAME  
SYNCHRONIZATION WORDS IN A MULTIDIMENSIONAL DIGITAL  
FRAME STRUCTURE"

Serial No. \_\_\_\_\_ filed \_\_\_\_\_ for "SYSTEM  
AND METHOD FOR PROGRAMMING THE VALUE OF FRAME  
15 SYNCHRONIZATION WORDS IN A MULTIDIMENSIONAL DIGITAL  
FRAME STRUCTURE"

Serial No. \_\_\_\_\_ filed \_\_\_\_\_ for "SYSTEM  
AND METHOD FOR PROGRAMMING THE BIT ERROR RATE OF  
FRAME SYNCHRONIZATION WORDS IN A MULTIDIMENSIONAL  
20 DIGITAL FRAME STRUCTURE"

Serial No. \_\_\_\_\_ filed \_\_\_\_\_ for "SYSTEM  
AND METHOD FOR PROGRAMMING THE QUANTITY OF FRAME  
SYNCHRONIZATION WORDS IN A MULTIDIMENSIONAL DIGITAL  
FRAME STRUCTURE"

25 Serial No. \_\_\_\_\_ filed \_\_\_\_\_ for "SYSTEM  
AND METHOD FOR SELECTIVELY BROADCASTING A  
MULTIDIMENSIONAL DIGITAL FRAME STRUCTURE"

Serial No. \_\_\_\_\_ filed \_\_\_\_\_ for "SYSTEM  
AND METHOD FOR PROGRAMMING LOSS OF SYNCHRONIZATION  
IN A MULTIDIMENSIONAL DIGITAL FRAME STRUCTURE"

Serial No. \_\_\_\_\_ filed \_\_\_\_\_ for "SYSTEM  
5 AND METHOD FOR TRANSLATING OVERHEAD BYTES IN A  
MULTIDIMENSIONAL DIGITAL FRAME STRUCTURE"

Serial No. 09/527,343, filed 17 March 2000 for  
"TRANSPOSABLE FRAME SYNCHRONIZATION STRUCTURE"

Serial No. 09/528,021, filed 17 March 2000 for  
10 "PROGRAMMABLE SYNCHRONIZATION STRUCTURE WITH  
AUXILIARY DATA LINK"

## BACKGROUND OF THE INVENTION

### 1. Field of the Invention

15 This invention generally relates to communications  
structured in a digital wrapper and, more particularly, to a system and  
method for programming the frame synchronization criteria in the  
communication frame structure.

### 2. Description of the Related Art

20 Frame synchronization is used in most communication  
technologies where extra bandwidth is required to add functionality into  
the original data stream. There is no standard practice with respect to  
the bandwidth allocation of frame synchronization words. Neither is  
there an ability to customize the use of that bandwidth for  
25 synchronization thresholding adaptive to various bit error rate conditions

in a variable rate forward error correction (FEC) system implementing an interleaved multi-frame superframe.

It would increase information bandwidth if all the bytes, or words of a digital frame structure could be devoted to payload. However a significant portion of the frame is devoted to FEC to remove errors that are introduced in data transmission lines. Even fiber optic systems can be noisy enough to introduce errors, when the distance between nodes is great enough. Part of the frame must also be used for maintenance functions between communication nodes. Further, a portion of the frame must be devoted to synchronization. The synchronization function becomes even more critical when the frames are scrambled to randomize the data density. These maintenance and synchronization bytes reside in the overhead section of the frame. However, as mentioned above, there is no standard protocol defining the number, location, or value of frame synchronization bytes. Hardware devoted to any particular protocol or implementation quickly becomes obsolete. An exclusively software approach to managing communications is cumbersome. . . .

It would be advantageous if communication hardware could be developed that permitted flexibility in defining the frame synchronization byte structure, so as to permit the hardware to be used in a number of communication standards.

It would be advantageous if the number of frames, with recognizable frame synchronization bytes (FSBs), required for synchronization could be made selectable in a multidimensional digital frame structure.

Likewise, it would be advantageous if the quantity, location, values, and bit error rates of the FSBs could be made selectable to accommodate a number of communication standards, or changes to existing standards.

5

## SUMMARY OF THE INVENTION

Accordingly, an integrated circuit communication device is provided that includes programmable features that permit a user to allocate bandwidth for frame synchronization words, and set up the error  
10 thresholding, adjusting the robustness to bit errors and the time required to achieve synchronization. More specifically, the device provides for:

- a programmable frame synchronization word bandwidth;
- programmable frame synchronization words;
- different synchronization words with independent allocated  
15 bandwidths;
- frame synchronization word translation between the decoder and encoder;
- frame synchronization word error thresholding for system robustness;
- 20 programmable synchronization (in-to-sync) parameters; and
- programmable out-of-sync synchronization parameters.

Broadly, this invention describes a structure for programming key features for a frame synchronization system using a multidimensional digital frame. More particularly, an example of the  
25 invention is provided in a programmable framing structure for a forward error correction (FEC) encoded channel interleaved to 16 levels with a 4

frame superframe. The structure provides the user with the ability to allocate the amount of bandwidth dedicated to the frame synchronization words, to control the speed of synchronization and the robustness to channel noise. As an added benefit, a programmable synchronization threshold can be set to even further increase the robustness to channel noise and to decrease the synchronization time under high bit error rate conditions. These techniques together allow users to increase network security and partitioning as well as to dynamically reallocate the bandwidth occupied for framing based on link conditions without corrupting user data.

Details of the programmable communication IC and a method for variably programming the frame synchronization structure in a multidimensional digital frame structure are provided below.

## BRIEF DESCRIPTION OF THE DRAWING

Fig. 1 is a schematic block diagram of a system for variably programming frame synchronization criteria in the communication of a multidimensional digital frame structure.

Fig. 2 is a diagram illustrating a multidimensional digital frame structure.

Fig. 3 is a flowchart depicting a method for variably programming the synchronization of a multidimensional digital frame structure.

Fig. 4 is a flowchart depicting a method for variably programming the synchronization of received frames in the communication of a multidimensional digital frame structure.

Fig. 5 is an example of a subframe consisting of a 255-byte RS(255,239) block with an overhead (OH) byte as the first byte, followed by 238 user payload bytes, and 16 bytes of RS parity.

Fig. 6 illustrates the stacking of subframes in a frame.

5

## DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Fig. 1 is a schematic block diagram of a system for variably programming frame synchronization criteria in the communication of a multidimensional digital frame structure. The system 100 comprises a transmitter 102 with a frame generator 104. The frame generator 104 includes an overhead generator 106 having an input to accept commands on line 108 for selecting the number of synchronization bytes in the overhead section of a frame. A payload generator 110 supplies the information that is to be transmitted. The information can be generated at the transmitter 102, or it can be the payload of a previously received frame. In this case, the transmitter 102 would be acting as a relay. The frame generator 104 also includes an FEC section, or encoder 112. The encoder codes the payload so that errors due to degradation can be removed at the destination. In some aspects of the invention, parts of the overhead section, or the entire overhead section is encoded, along with the payload. Commands to select FSB locations are accepted on line 114, and commands to select FSB values are accepted on line 116.

Fig. 2 is a diagram illustrating a multidimensional digital frame structure. As can be seen from examining Frame 1, each frame is considered to be multidimensional because it includes a plurality of rows,

where each row includes an overhead byte, payload bytes, and parity (FEC) bytes. The present invention is not limited to any particular number of bytes to the overhead, payload, and FEC sections, or to any particular number of rows.

5               Returning to Fig. 1, the system 100 includes a receiver 120 with a frame receiver 122. The frame receiver 122 includes an overhead receiver 124 with an input on line 126 to accept commands for selecting the quantity of frame synchronization bytes used for synchronizing the received frame. The overhead receiver 124 synchronizes the frame in  
10    response to recognizing the selected frame synchronization bytes. In other aspects of the invention, the frame synchronization function is performed by other elements of the receiver 120 (not shown). Once the FSBs have been identified, synchronization is possible. The frame structure is predetermined and it becomes possible for the receiver 120 to  
15    identify overhead, payload, and FEC sections. FSBs act as a known pilot symbol in a stream of uncertain data. Once the structure is identified, the frame can be decoded and the payload corrected. In some aspects of the invention, the receiver 120 is part of a relay, and the payload is relayed as is, or modified before it is retransmitted. Commands to accept FSB bit  
20    error rates are accepted on line 128, commands to accept FSB locations are accepted on line 130, and commands to accept FSB values are accepted on line 134. Commands to accept synchronization and non-synchronization information are accepted on lines 138 and 140, respectively.

25               The overhead receiver 124 has an input on line 138 to accept commands for selecting a number of frames, with recognized frame

synchronization bytes, required for synchronization. The overhead receiver 134 synchronizes the frame in response to the selected number of frames with recognized FSBs.

The frame receiver 122 defines a superframe structure with a  
5 predetermined number of frames per superframe, and the overhead  
receiver 124 selects a number of frames in each superframe. The frame  
receiver 124 synchronizes received frames in response to the selected  
number of frames in each superframe. For example, the frame receiver  
122 defines a first and a second frame in the superframe, and the  
10 overhead receiver 124 selects frames from the group including the first  
frame and the second frame. Thus, not all frames in a superframe need be  
used for synchronization.

In some aspects of the invention, the frame receiver 122  
defines a first frame in a first superframe and a first frame in a second  
15 superframe. Then, the overhead receiver 124 selects frames from the  
group including the first frame in the first superframe and the first frame  
in the second superframe. If the first frame of each superframe includes  
recognized FSBs, synchronization can occur.

The frame receiver 122 defines a first plurality of frames in  
20 each superframe, and the overhead receiver 124 selects frames from the  
first plurality of frames in each superframe. More specifically, the frame  
receiver 122 defines a second plurality of superframes. Then, the  
overhead receiver 124 selects a number of consecutive superframes having  
frames with recognized frame synchronization bytes. That is, the frame  
25 receiver 122 selects a first number of frames with recognized frame  
synchronization bytes in a second number of consecutive superframes.



For example, the frame receiver 122 defines a superframe having four frames. If the overhead receiver 124 recognizes frame synchronization bytes in the first frame for two consecutive superframes, synchronization occurs.

5           The recognition of FSBs is based upon a variety of selectable criteria in the present invention. Most of the criteria needs to be agreed upon by the receiver 120 and the transmitter 102. The overhead receiver 124 selects the value of frame synchronization bytes in the frame overhead section. The overhead receiver 124 recognizes frame  
10 synchronization bytes in the frame overhead sections in response to recognizing the value of the frame synchronization bytes.

Typically, the FSB word is a byte of eight bits, although the invention is not limited to an FSB word of any particular length. With eight bit values,  $2^8$  possible FSB values are possible for every FSB.  
15 Likewise, the FSB values may vary inside a frame. If a frame includes a first plurality, i.e., sixteen, overhead bytes, and sixteen FSBs are selected, then each of the sixteen FSBs may be the same byte value. On the other extreme, each FSB byte value may be different, and sixteen FSB values can be used in the frame. Thus, each frame can be synchronized using  
20 FSBs having different values.

In one aspect of the invention, the overhead receiver 124 selects first frame synchronization bytes having a first value and second frame synchronization bytes having a second value. Then, the overhead receiver 124 synchronizes the received frame in response to recognizing  
25 the first frame synchronization bytes having the first value and the second frame synchronization bytes having the second value. Although an

example using two different FSB values is presented above, it is possible to have as many FSBs as there are rows in a frame. Thus, in the frame structure of Fig. 3, each frame could have a first plurality (sixteen in the example of Fig. 2) of unique FSB values. Even more FSB values could be used if a greater portion of each row was devoted to the overhead section, at the expense of the payload or FEC sections.

Likewise, the value of the FSBs can change between frames. In some aspects of the invention, the overhead receiver 124 selects a first number of frame synchronization bytes having a first value in a first frame and a second number of frame synchronization bytes having a second value in a second frame. The overhead receiver synchronizes the received frame in response to recognizing the first number of frame synchronization bytes having the first value in the first frame and the second number of frame synchronization bytes having the second value in the second frame.

Alternately, multiple FSB values can be used in multiple frames. For example, the overhead receiver 124 selects a first number of frame synchronization bytes having a first value and a second number of frame synchronization bytes having a second value in a first and second frame. Then, the overhead receiver 124 synchronizes the frame in response to recognizing frame synchronization bytes having the first and second values in the first and second frames. In some aspects of the invention, the first FSB values are located in a first frame, and the second FSB values are located in a second frame.

In a simple aspect of the invention, the overhead generator 106 selects frame synchronization bytes having a first value, in response

to commands on line 136, and the overhead receiver 124 selects frame synchronization bytes having the first value. However, the overhead receiver 124 need not select all the FSB values supplied by the overhead generator. For example, the overhead generator 106 selects a first  
5 number of frame synchronization bytes having a first value in a first number of locations and a second number of frame synchronization bytes in a second number of locations having a second value. The overhead receiver 124 selects a third number of frame synchronization bytes in a third number of locations, less than the first number, having the first  
10 value, and a fourth number of frame synchronization bytes in a fourth number of locations, less than the second number, having the second value. The overhead receiver 124 synchronizes the received frame in response to recognizing the third number of frame synchronization bytes having the first value in the third number of locations, and the fourth  
15 number of frame synchronization bytes having the second value in the fourth number of locations.

In some aspects of the invention, the overhead receiver 124 selects the number of frame synchronization bytes in the frame overhead section. The overhead receiver 124 recognizes frame synchronization  
20 bytes in the frame overhead sections in response to recognizing the number of the frame synchronization bytes.

For example, the overhead receiver 124 selects a first number of byte values for a first frame of the superframe, and synchronizes the first frame by recognizing the first number of byte values  
25 in the first frame of the superframe.

In a simple aspect of the invention, the overhead generator 106 selects a first number of frame synchronization byte values in the overhead section of a frame, and the overhead receiver 124 selects the first number of frame synchronization byte values for recognition for the first frame. However, the overhead receiver need not select the all the FSBs that have been supplied by the overhead generator 106. For example, when the overhead generator supplies a first number of frame synchronization byte values for a first frame, the overhead receiver 124 can select a second number of byte values, less than the first number, for the first frame. The overhead receiver synchronizes the received frame in response to recognizing the second number of frame synchronization byte values in the first frame.

Further, the overhead generator 106 can supply a first number of FSBs in a first frame and a second number of FSBs in the second frame. The overhead receiver 124 can synchronize using the first number of FSB values in the first frame and the second number of FSBs in the second frame. However, as explained, the overhead receiver 124 is not required to use all the FSBs that are generated.

In some aspects, the overhead receiver 124 selects the bit error rate for the frame synchronization bytes, so that the overhead receiver 124 recognizes frame synchronization bytes having a bit error rate less than, or equal to, the selected bit error rate. Alternately, an average FSB error rate is used for each selected frame. Then, the average FSB BER in each frame is used to determine if the FSBs in that frame have been recognized. In some aspects, the average BER applies to all

frames. In other aspects, the average BER is selected on a frame-by-frame basis.

A byte can be programmed and entered into the receiver 120 for selecting the BER. For example, the overhead receiver 124 can select  
5 a BER in the range from zero to a second plurality of bits, where the programmable byte includes a second plurality of bits. Note that the BER can be set for different values in recognizing FSBs in a single frame. Likewise, the BER can be set for different values between frames, or between superframes. The BER can be made to correspond to the FSB  
10 location and/or the FSB value.

In some aspects of the invention, the overhead receiver 124 selects the location of the frame synchronization bytes in the frame overhead section. The overhead receiver 124 recognizes frame synchronization bytes in the frame overhead sections in response to  
15 recognizing the location of the frame synchronization bytes.

For example, the overhead receiver 124 selects a first number of byte locations, and synchronizes the received frame in response to recognizing frame synchronization bytes in the first number of selected locations.

20 However, the frame locations can vary between frames, and between superframes. For example, the overhead receiver 124 selects a first number of FSB byte values in a first number of frames and a second number of FSB byte values in a second number of locations. Likewise, a first number of locations can be selected in a first frame of the  
25 superframe, and a second number of locations in a second frame. The overhead receiver 124 synchronizes the received frame in response to

recognizing frame synchronization bytes in the first number of selected locations in the first frame and the second number of selected locations in the second frame. Looking at Fig. 2 momentarily to refine the above-mentioned example, locations OH 1, OH 2, and OH 3 can be selected for the first frame, while locations OH 5, OH 6, and OH 7 can be selected for the second frame. This example would, of course, require the overhead generator 106 to supply FSBs in at least the above-mentioned byte locations.

In a simple aspect of the invention, the overhead generator 106 selects a first number of locations for a first number of frame synchronization bytes, in response to commands on line 132, and the overhead receiver 124 selects the first number of locations for the first number of frame synchronization bytes. However, the overhead receiver need not select all the locations provided by the overhead generator 106. For example, the overhead generator 106 selects a first number of locations for a first number of frame synchronization bytes, and the overhead receiver 124 selects a second number of locations for a second number of frame synchronization bytes, less than the first number, and synchronizes the received frame in response to recognizing frame synchronization bytes in the second number of selected locations. Momentarily examining Fig. 2, the overhead generator 106 may supply FSBs in locations OH 1 through OH 6, while the overhead receiver selects locations OH 4 through OH 6 for use.

When the overhead receiver is not able to recognize FSBs, synchronization is lost. Once again, the way in which the system loses synchronization is programmable. The overhead receiver 124 accepts

commands on line 140 for selecting a number of consecutive non-recognized frames, or missed frames, where a missed frame includes non-recognized FSBs. Then, the overhead receiver 124 falls out of synchronizing in response to the selected number of consecutively non-recognized frames.

Fig. 3 is a flowchart depicting a method for variably programming the synchronization of a multidimensional digital frame structure. Although the method is depicted as a sequence of numbered steps for clarity, no order should be inferred from the numbering unless explicitly stated. The method begins with Step 200. Step 202 defines a frame with an overhead section having a predetermined number of bytes. Step 204 recognizes frame synchronization bytes in the frame overhead sections. Step 206 selects a number of frames, with recognized frame synchronization bytes, required for synchronization. Step 208 synchronizes received frames in response to the selected number of frames with recognized frame synchronization bytes.

Step 201 defines a superframe structure with a predetermined number of frames per superframe. Selecting a number of frames, with recognized frame synchronization bytes, required for synchronization in Step 206 includes selecting a number of frames in each superframe. Synchronizing received frames in response to the selected number of frames in Step 208 includes synchronizing received frames in response to the selected number of frames in each superframe.

In some aspects of the invention, defining a superframe structure with a predetermined number of frames per superframe in Step 201 includes defining a first and a second frame in the superframe.

Selecting a number of frames, with recognized frame synchronization bytes, required for synchronization in Step 206 includes selecting frames from the group including the first frame and the second frame.

In some aspects of the invention, defining a superframe structure with a predetermined number of frames per superframe in Step 5 201 includes defining a first frame in a first superframe and a first frame in a second superframe. Selecting a number of frames, with recognized frame synchronization bytes, required for synchronization in Step 206 includes selecting frames from the group including the first frame in the 10 first superframe and the first frame in the second superframe.

In some aspects, defining a superframe structure with a predetermined number of frames per superframe in Step 201 includes defining a first plurality of frames in each superframe. Selecting a number of frames, with recognized frame synchronization bytes, required 15 for synchronization in Step 206 includes selecting frames from the first plurality of frames in each superframe.

In some aspects of the invention, defining a superframe structure with a predetermined number of frames per superframe in Step 201 includes defining a second plurality of superframes. Selecting a 20 number of frames, with recognized frame synchronization bytes, required for synchronization in Step 206 includes selecting a number of consecutive superframes having frames with recognized frame synchronization bytes.

In some aspects of the invention, selecting a number of frames, with recognized frame synchronization bytes, required for 25 synchronization in Step 206 includes selecting a first number frames with



recognized frame synchronization bytes in a second number of consecutive superframes.

In some aspects of the invention, defining a superframe structure with a predetermined number of frames per superframe in Step  
5 201 includes defining a superframe having four frames. Selecting a number of frames, with recognized frame synchronization bytes, required for synchronization in Step 206 includes recognizing frame synchronization bytes in the first frame for two consecutive superframes.

In some aspects, Step 203a selects the value of frame  
10 synchronization bytes in the frame overhead section. Recognizing frame synchronization bytes in the frame overhead sections in Step 204 includes recognizing the value of the frame synchronization bytes.

In some aspects of the invention, defining a frame in Step  
15 202 includes defining each frame synchronization byte having a second plurality of bits. Selecting the value of the frame synchronization bytes in Step 203a includes selecting a second plurality of bits for each frame synchronization byte.

In some aspects, Step 203b selects the quantity (number) of frame synchronization bytes in the frame overhead section. Defining the  
20 frame in Step 202 includes defining the overhead section having a first plurality of bytes. Selecting the number of frame synchronization bytes in the overhead section in Step 203b includes selecting a first number of bytes in the range from zero to the first plurality of bytes. In some aspects of the invention, recognizing frame synchronization bytes in the  
25 frame overhead sections in Step 204 includes recognizing the number of the frame synchronization bytes.

Some aspects of the invention have a further step. Step 203c selects the bit error rate for the frame synchronization bytes. Recognizing frame synchronization bytes in the frame overhead sections in Step 204 includes recognizing the frame synchronization bytes having a bit error  
5 rate less than, or equal to, the selected bit error rate.

In some aspects, selecting a frame synchronization byte bit error rate in Step 203c includes selecting an average BER for each selected frame. Then, FSBs are recognized in the frame, in Step 204, when the average FSB BER is at least equal to the selected BER. In some  
10 aspects, a programmable bytes is used to select a BER in the range from zero to a second plurality of errors, where programmable byte includes a second plurality of bits.

Step 203d selects the location of the frame synchronization bytes in the frame overhead section. Recognizing frame synchronization  
15 bytes in the frame overhead sections in Step 204 includes recognizing the location of the frame synchronization bytes.

Step 210 selects a number of frames, with non-recognized frame synchronization bytes, required for falling out of synchronization. Step 212 falls out of synchronizing in response to the selected number of  
20 frames with non-recognized frame synchronization bytes.

Fig. 4 is a flowchart depicting a method for variably programming the synchronization of received frames in the communication of a multidimensional digital frame structure. The method begins with Step 300. Step 302 defines a frame with an overhead  
25 section having a predetermined number of bytes. Step 304 sends a frame. Step 306 receives the frame. Step 308 recognizes frame synchronization

bytes in the frame overhead sections. Step 310 selects a number of frames, with recognized frame synchronization bytes, required for synchronization. Step 312 synchronizes received frames in response to the selected number of frames with recognized frame synchronization  
5 bytes.

An example of the above, broadly-stated invention is presented below. The typical frame structure uses a 255-byte FEC code as the basic building block which is referred to as a subframe, or row. Each subframe is defined by a Reed-Solomon (RS) codec that is intended to  
10 provide 8 byte correction capability over the 255-byte block. This codec is referred to as RS(255,239) for 8 byte correction.

Fig. 5 is an example of a subframe consisting of a 255-byte RS(255,239) block with an overhead (OH) byte as the first byte, followed by 238 user payload bytes, and 16 bytes of RS parity. The OH byte may  
15 be used as a Frame Synchronization Byte (FSB) or other programmable function.

Fig. 6 illustrates the stacking of subframes in a frame. The subframes are stacked in a structure that is 16 deep to create a frame as shown in Fig. 2. The bytes are transmitted in an interleaved fashion from  
20 top to bottom and left to right. Four frames are used to create a superframe as shown in Fig. 2. A superframe contains 64 overhead locations. The frames are transmitted in order from frame 1 to frame 4.

The first frame in the superframe is the one that traditionally contains frame synchronization bytes. The user may select  
25 to have two different FSB types (FSB1 and FSB2). The ability to have two different FSBs and the ability to program them with custom values

provide the user added control over their system both in terms of resistance to false synchronization and the ability to distinguish different data sources from each other. This is useful when there are multiple wavelengths that can be received, all carrying the same frame structure, but only one of which is allowed/permissioned to be received at a particular node. Table 1 provides an example of how the FSB types are defined.

Table 1. Frame Synchronization Byte Definitions

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FSB 2								FSB 1							
Mode	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
Default	0	0	1	0	1	0	0	0	1	1	1	1	0	1	1	0

Bit Positions	Function	Description
15:8	FSB 2	Frame Synchronization Byte 2 (Default value = 28h)
7:0	FSB 1	Frame Synchronization Byte 1 (Default value = F6h)

Accommodations are made for two separate frame synchronization bytes.

The FSBs are also programmable in number. In Frame 1, OH 1 to OH 16 can be programmed to be FSB1, FSB2, or a non-FSB function. This expands on the variability afforded by having two different FSBs. By using more of the available bandwidth for FSBs, the user decreases the chance of false synchronization and in general, decreases the amount of time required to acquire synchronization. The user can customize this parameter to optimize for specific requirements and link conditions.

Table 2 demonstrates the structure required to indicate which OH bytes are used for FSBs and Table 3 illustrates the structure that sets whether to use FSB1 or FSB2. The FSB programmability exists

independently in both the transmit and receive portions of the device. In the receive section of the device, the number of recognized consecutive FSB groups required to declare synchronization is programmable, as well as the number of bad consecutive FSB groups required to lose  
5 synchronization. This is the first level of thresholding for adjusting the tolerance to bit error rate and the amount of time required to declare synchronization. Table 4 is an example of synchronization parameter setting. The fewer consecutive FSB groups required to achieve synchronization, the shorter the acquisition time. The more consecutive  
10 bad FSB groups required to lose synchronization, the greater the system tolerant of higher bit error rates.

The second level of thresholding is to define what is considered a good group of FSBs vs. a bad group of FSBs. This is done by specifying the number of errors that are allowed within a group of FSBs  
15 and still be considered good as shown in Table 4. This feature has the advantage of decreasing synchronization time in the presence of high bit error rates. Longer FSB groups are permitted which decrease the probability of false synchronization, without incurring the penalty of difficult synchronization in the presence of noise. Further, the FEC code  
20 is given greater opportunity to correct errors.

Another benefit of the second level of thresholding is the ability to dynamically customize the values of the FSBs. This thresholding provides network security, and dynamically allocates more bytes for synchronization based on the link conditions, without causing a  
25 resynchronization to occur.

Table 2. Frame Synchronization Byte Locations Register

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OH #1	OH #2	OH #3	OH #4	OH #5	OH #6	OH #7	OH #8	OH #9	OH #10	OH #11	OH #12	OH #13	OH #14	OH #15	OH #16
Mode	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
Default	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0

Bit Positions	Function	Description
15	OH#1	0: Not a Frame Synchronization Byte 1: Frame Synchronization Byte (FSB) (Default)
14	OH#2	0: Not a Frame Synchronization Byte 1: Frame Synchronization Byte (FSB) (Default)
13	OH#3	0: Not a Frame Synchronization Byte 1: Frame Synchronization Byte (FSB) (Default)
12	OH#4	0: Not a Frame Synchronization Byte 1: Frame Synchronization Byte (FSB) (Default)
11	OH#5	0: Not a Frame Synchronization Byte 1: Frame Synchronization Byte (FSB) (Default)
10	OH#6	0: Not a Frame Synchronization Byte 1: Frame Synchronization Byte (FSB) (Default)
9	OH#7	0: Not a Frame Synchronization Byte (Default) 1: Frame Synchronization Byte (FSB)
8	OH#8	0: Not a Frame Synchronization Byte (Default) 1: Frame Synchronization Byte (FSB)
7	OH#9	0: Not a Frame Synchronization Byte (Default) 1: Frame Synchronization Byte (FSB)
6	OH#10	0: Not a Frame Synchronization Byte (Default) 1: Frame Synchronization Byte (FSB)
5	OH#11	0: Not a Frame Synchronization Byte (Default) 1: Frame Synchronization Byte (FSB)
4	OH#12	0: Not a Frame Synchronization Byte (Default) 1: Frame Synchronization Byte (FSB)
3	OH#13	0: Not a Frame Synchronization Byte (Default) 1: Frame Synchronization Byte (FSB)
2	OH#14	0: Not a Frame Synchronization Byte (Default) 1: Frame Synchronization Byte (FSB)
1	OH#15	0: Not a Frame Synchronization Byte (Default) 1: Frame Synchronization Byte (FSB)
0	OH#16	0: Not a Frame Synchronization Byte (Default) 1: Frame Synchronization Byte (FSB)

This table defines which of the overhead bytes (1-16) are going to be used as FSBs.

5

10

Table 3. Frame Synchronization Byte Types

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OH #1	OH #2	OH #3	OH #4	OH #5	OH #6	OH #7	OH #8	OH #9	OH #10	OH #11	OH #12	OH #13	OH #14	OH #15	OH #16
Mode	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
Default	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0

**NOTE:** If an OH byte is not defined to be an FSB in the Decoder Frame Synchronization Byte Locations Register, the corresponding bit in this register has no significance.

Bit Positions	Function	Description
15	OH#1	0: FSB 1 (Default) 1: FSB 2
14	OH#2	0: FSB 1 (Default) 1: FSB 2
13	OH#3	0: FSB 1 (Default) 1: FSB 2
12	OH#4	0: FSB 1 1: FSB 2 (Default)
11	OH#5	0: FSB 1 1: FSB 2 (Default)
10	OH#6	0: FSB 1 1: FSB 2 (Default)
9	OH#7	0: FSB 1 (Default) 1: FSB 2
8	OH#8	0: FSB 1 (Default) 1: FSB 2
7	OH#9	0: FSB 1 (Default) 1: FSB 2
6	OH#10	0: FSB 1 (Default) 1: FSB 2
5	OH#11	0: FSB 1 (Default) 1: FSB 2
4	OH#12	0: FSB 1 (Default) 1: FSB 2
3	OH#13	0: FSB 1 (Default) 1: FSB 2
2	OH#14	0: FSB 1 (Default) 1: FSB 2
1	OH#15	0: FSB 1 (Default) 1: FSB 2
0	OH#16	0: FSB 1 (Default) 1: FSB 2

Table 4. Decoder Framing Controls

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Check to go In-Frame				Check to go Out-of-Frame				Unused		FSB Errors Allowed					
Mode	rw	rw	rw	rw	rw	rw	rw	rw	ro	ro	rw	rw	rw	rw	rw	rw
Default	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0

Bit Positions	Function	Description
15:12	Check to go In-Frame	Number of correct consecutive FSB groups required to go In-Frame. (Default value = 2)
11:8	Check to go Out-of-Frame	Number of incorrect consecutive FSB groups required to go Out-of-Frame. (Default value = 4)
7:5	Unused	
5:0	FSB Errors Allowed	The total number of erred bits allowed in a group of FSBs to still consider a match. (Default value = 8). This number is used regardless of the synchronization state (In-Frame or Out-of-Frame).

Table 4 sets the synchronization thresholding parameters.

- 5                   A system and method have been provided that illustrate the advantages of providing programmable synchronization criteria. The invention can fully customize the synchronization methodology, to optimize robustness to noise and synchronization time. The invention specifically illustrates an example using different FSB values,
- 10   programmable numbers of FSBs, programmable gain synchronization, programmable lose synchronization parameters, and programmable bit error tolerance within the FSBs themselves. However, the invention is not limited to an particular gain synch algorithm, or any specific framing structure. The invention permits users to create different synchronization
- 15   criteria based upon circumstances and quality of the received signal. Other variations and embodiments of the invention will occur to those skilled in the art.

WE CLAIM: